Сложность проблемы «Выполнимость» для бинарных программ

Мубаракзянов Рустам Гамирович

Казанский (Приволжский) федереальный университет, e-mail: rustam.mubarakzjanov@ekpfu.ru

Вычислительная модель, называемая бинарной программой, хорошо известна [1]. Напомним основные определения.

Детерминированной бинарной программой (BP) P на множестве переменных $X=\{x_0, ..., x_{n-1}\}$ называется ориентированный ациклический граф с одной начальной вершиной и двумя финальными вершинами. Одна из финальных вершин помечена 0, другая — 1 (назовем ее принимающей). Каждая нефинальная вершина этого графа помечена переменной из X, и из нее выходят ровно две дуги, помеченные 0 и 1. Процесс вычисления детерминированной BP при фиксировании значений переменных из X сводится к следующему. Вычисление начинается в корне s программы. Если переменная, которой помечен s, имеет значение $a \in \{0, 1\}$, то осуществляется переход в сына s по дуге, помеченной a. Из этой вершины переход осуществляется в зависимости от значения помечающей ее переменной. Так как путь, соответствующий значениям переменных, выбирается однозначно, каждому набору значений переменных можно сопоставить пометку той финальной вершины, в которую приводит вычисление. Эта пометка является значением вычисляемой функции.

Если в определении бинарной программы добавить возможность наличия недетерминированных вершин, т. е. не помеченных никакой переменной, или вероятностных вершин, т. е. помеченных случайными переменными, принимающими значения 0 и 1 с вероятностью 1/2, получим определения недетерминированной и вероятностной программы соответственно. Вероятностная программа P определяет функцию $c_P : \{0, 1\}^n \to [0, 1]$, $c_P(x)$ — вероятность того, что P, имея на входе x, достигает принимающую вершину. Программа P вычисляет функцию h, если $c_P(x) > 1/2$ для $h(x) = 1$, и $c_P(x) < 1/2$ для $h(x) = 0$, т. е. вероятность, что P выдает $h(x)$, больше 1/2 для любого набора значений переменных x. Если эта вероятность больше $1 - \varepsilon$ для некоторого ε, $0 \leq \varepsilon < 1/2$, то вычисление называется вычислением с ограничением на ошибку (ε — ошибка вычисления).

Определим сложность программы P как число ее вершин. В целях получения соотношений между различными классами сложности рассматриваются ограниченные классы бинарных программ [1]. k раз читающая бинарная программа (BPk) — бинарная программа, которая на любом пути вычисления каждую переменную читает не более k раз. Один раз читающая BP ($BP1$) называется упорядоченной (обозначается $OBDD$), если переменные читаются в определенном порядке. $OBDD$ интересны в связи с возможностью их практического использования [2]. Программа BPk называется $kOBDD$, если ее можно разбить на k слоев так, что переменные в каждом слое читаются не более одного раза в соответствии с одним и тем же порядком.

Обозначим классы функций, вычислимых детерминированными, недетерминированными, вероятностными с ограничением на ошибку и вероятностными общего вида бинарными программами полиномиальной сложности $P{-}BP$, $NP{-}BP$, $BPP{-}BP$ и $PP{-}BP$ соответственно. Для ограниченных классов бинарных программ в обозначениях классов сложности суффикс "$-BP$" заменим соответствующим сокращением. Известны соотношения между классами сложности для $OBDD$ [3, 4].

Классы бинарных программ полиномиальной сложности будем обозначать так же, как и соответствующие классы сложности. Например, вероятностные (без ограничения на ошибку) $OBDD$ полиномиального размера обозначим как $PP{-}OBDD$. Из контекста будет ясно, для чего используются обозначения: для бинарных программ или для класса сложности.

Назовем бинарную программу уровневой, если для любой ее нефинальной вершины v все пути от корня до v имеют одинаковую длину. Такая программа может быть разбита на уровни: каждый уровень содержит нефинальные вершины, одинаково удаленные от корня. Шириной уровневой программы называется максимальное число вершин на одном уровне. Если рассматривать BP, ширина которых ограничена некоторой функцией $\omega(n)$, обозначим их $BP^{\omega(n)}$.

При обобщениях $P{-}OBDD$ важно, чтобы полученные бинарные программы сохраняли свойства, позволяющие эффективно работать с ними. Важной проблемой в теории сложности является сложность задачи «Выполнимость».

Для BP полиномиального размера в классе $NP{-}BP1$ тест на «Выполнимость» полиномиален. Следовательно, аналогичное утверждение верно $P{-}BP1$, $NP{-}OBDD1$. Этот тест осуществляет проход по графу программы в глубину. Так как каждая переменная читается не более одного раза, можно не учитывать значения переменных. Если рассматривать BP константной ширины, то ограничение на полиномиальный размер выполнится.

Обобщим рассмотренные классы сложности.

Класс $P{-}BP1$ строго включается в $P{-}BP2$. Для последнего класса, даже для $P{-}2OBBDD^3$, проблема «Выполнимость» NP-полна [5].

Теорема. *Для любого целого k и для любой целочисленной функции $\omega(n)$ справедливо следующее $PP{-}kOBDD^{\omega(n)} \subseteq PP{-}OBDD^{\omega(n)^{2k}}$.*

Идея доказательства состоит в замене наборов состояний уровней исходной $kOBDD$ всех k слоев, находящихся на равном удалении от начала слоя, на один уровень состояний $OBDD$.

Проблема «Выполнимость» NP-полна для $BPP{-}OBDD$ [6] и, следовательно, для $PP{-}OBDD$.

Список литературы

[1] Wegener I. Branching programs and binary desicion diagrams: theory and applications. — SIAM monographs on discrete mathematics and applications. — Philadelphia, USA, 2000. — 408 P.

[2] Bryant R. E. Graph-based algorithms for Boolean function manipulation // IEEE Transact. on Comput. — 1986. — V. 35. — P. 677–691.

[3] Ablayev F. Randomization and nondeterminism are incomparable for ordered read-once branching programs // Proc. ICALP'97, LNCS 1256, Springer. — 1997. — P. 195–202.

[4] Ablayev F., Karpinski M., Mubarakzjanov R. On BPP versus $NP \cup coNP$ for ordered read-once branching programs // Theoret. Comput. Sci. — 2001. — V. 264. — P. 127–137.

[5] Meinel Ch., Theobald T. Algorithms and data structures in VLSI design. — Berlin. : Springer. — 1998. — 267 P.

[6] Agrawal M., Thierauf T. The satisfiability problem for probabilistic ordered branching programs // Theory Comput. Systems (TOCS). — 2001. — V. 34. — P. 471–487.